<center>**REMARKS**</center>

Amendments to the specification have been made. The specification amendments revert the specification back to its content as originally filed. Support for the specification amendments can be found in the corresponding, originally filed paragraphs. No new matter has been added by these amendments. Claims 1, 30, 32, and 39 are amended. Claims 2-5, 7-25, 27-28, 31, 33, 36-37, and 40-48 are canceled. Support for the claims amendments can be found on pages 26-32, and in Figure 5 of the original as-filed application.

I.   Double Patenting

I.A.   Claims 1, 32, and 41

The Examiner has provisionally rejected claims 1, 32, and 41 under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 6, 21, and 25 of co-pending Application No. 10/675,777 (hereinafter '777). Claim 41 is canceled, making the rejection thereof moot.

Applicants acknowledge that this is a provisional rejection. As of the date of this response, neither the rejected claims, nor the claims of the co-pending application have been allowed. Therefore a terminal disclaimer would be premature at this point. Should the Examiner indicate that either the rejected claims, or the reference be allowable, a terminal disclaimer will be filed at that time.

I.B.   Claims 1, 32, and 41

The Examiner has provisionally rejected claims 1, 32, and 41 under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 2, 10, and 20 of co-pending Application No. 10/675,778 (hereinafter '778). Claim 41 is canceled, making the rejection thereof moot.

Applicants acknowledge that this is a provisional rejection. As of the date of this response, neither the rejected claims, nor the claims of the co-pending application have been allowed. Therefore a terminal disclaimer would be premature at this point. Should the Examiner indicate that either the rejected claims, or the reference be allowable, a terminal disclaimer will be filed at that time.

I.C.   Claims 1, 32, and 41

The Examiner has provisionally rejected claims 1, 32, and 41 under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 3 and 17 of co-pending Application No. 10/675,872 (hereinafter '872). Claim 41 is canceled, making the rejection thereof moot.

Applicants acknowledge that this is a provisional rejection. As of the date of this response, neither the rejected claims, nor the claims of the co-pending application have been allowed. Therefore a terminal disclaimer would be premature at this point. Should the Examiner indicate that either the rejected claims, or the reference be allowable, a terminal disclaimer will be filed at that time.

I.D.   Claims 6, 34, and 43

The Examiner has provisionally rejected claims 6, 34, and 43 under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 4, 12, and 20 of co-pending Application No. 10/675,721 (hereinafter '721). Claim 43 is canceled, making the rejection thereof moot.

Applicants acknowledge that this is a provisional rejection. As of the date of this response, neither the rejected claims, nor the claims of the co-pending application have been allowed. Therefore a terminal disclaimer would be premature at this point. Should the Examiner indicate that either the rejected claims, or the reference be allowable, a terminal disclaimer will be filed at that time.

I.E.   Claims 1, 32, and 41

The Examiner has provisionally rejected claims 1, 32, and 41 under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1, 12, and 23 of co-pending Application No. 10/682,385 (hereinafter '385). Claim 41 is canceled, making the rejection thereof moot.

Applicants acknowledge that this is a provisional rejection. As of the date of this response, neither the rejected claims, nor the claims of the co-pending application have been allowed. Therefore a terminal disclaimer would be premature at this point. Should the Examiner indicate that either the rejected claims, or the reference be allowable, a terminal disclaimer will be filed at that time.

II.   Specification

The Examiner has objected to the Specification under 35 USC 312(a) as introducing new matter into the disclosure. The Applicants have amended the specification, reverting it to the originally filed format. In light of these amendments, the rejection is now moot.

III.   35 U.S.C. § 103, Obviousness

The Final Office Action has rejected claims 1-2, 6, and 26-48 under 35 U.S.C. § 103 as being unpatentable over *Gover et al.*, U.S. Patent No. 5,752,062, (hereinafter "*Gover*") in view of APA (Admitted Prior Art: Specifications: Description of Related Art, Pg. 2-3). This rejection is respectfully traversed.

Regarding this rejection, the Final Office Action states:

> As **per claim 1,** *Gover* discloses a method in a data processing system for monitoring execution of instructions, the method comprising: receiving a bundle, the bundle containing an instruction (e.g. Fig. 1; Fig. 3); responsive to receiving the bundle, determining whether the bundle contains an indicator, wherein the indicator (see col. 7, lines 15 to col. 7, lines 17) identifies the instruction as one that is to be monitored by a performance monitor unit (Fig. 5); responsive to a determination that the bundle contains the indicator, incrementing a counter (e.g. Fig. 5; col. 7, line 62 to col. 8, line 17) associated with the instruction wherein the incrementing providing provides a count of a number of times the instruction is executed (see Fig. 6a, 6b); and sending the bundle to a functional unit for execution of the instruction (col. 4, lines 44-49). *Gover* does not explicitly disclose 'instruction cache' unit that *receives the bundle, determine the instruction to be counted based on the indicator for monitoring, and sending the bundle from the ICU to the execution unit;* that is, not implementing the instruction cache, the sequence unit (see Fig. 1) and the monitoring unit (Fig. 2; Fig. 4; col. 7, lines 15 to col. 7, lines 17) in a combined functional unit such as a single instruction cache unit (ICU). *Gover* discloses bus interface between instruction cache and sequencer unit (see Fig. 1; col. 6, lines 7-20); sequencer unit depending on *rename buffer* interface (col. 6, lines 13-19); dispatching process including associating completion/allocation interfaces (e.g. Fig. 3) for updating information (or indicators) in a reorder buffer in terms of conditions (see *finished, exception -* col. 6, line 66, to col 7, line 2) based on which some monitoring action (e.g. condition 2 - col. 10, line 9-12; dispatch logic 74 - Fig. 2; col. 7, lines 15 to col. 7, lines 17) can be applied; that is, using the performance monitor unit, in conjunction with the special registers or MMCRn (e.g. Fig. 4; Fig. 5; Fig. 6a). The meta-information being dispatched from the bundle of instructions coming from the reorder buffer, sequencer, rename

buffer, and the tight relationship thereof with the original instruction cache (Fig. 1) and the performance monitor by *Gover* entails that the dispatched bundle contains instructions or data back and forth between cache and hardware monitoring tool. APA teaches combining hardware performance tools (Specifications, pg. 3) into a software application performance system or a trace tool using profiling. **In** view of the role played by the MMCRn and the sequencer, the cache interface unit, a rename process and the handling of runtime exception based on the hardware monitoring role as in *Gover*, i.e. the interdependency of functionality (hardware and software) units involved, it would have been obvious for one skill in the art at the time the invention was made to implement *Gover's* instruction cache, sequencer unit and monitoring unit *(performance monitor 50,* Fig. 4 -i.e. hardware tools) as one functionality called *instruction processing unit* -- or more arbitrarily a 'instruction cache', or ICU- to effectuate a performance monitoring/support functionality such as contemplated by APA in terms of faster hardware performance. That is, one would be motivated to do so because this ICU can be called upon to **receive bundle,** trigger **monitoring event** based on indicators set forth by the reorder buffer inside the sequencer unit and provide expedite monitoring action (based on hardware support) operating on data from the dispatched bundle and based on the dynamic condition/state (e.g. *exception, finished, completion)* indicated by the bundle as set forth above (Fig. 6a, 6b) then accordingly **send the bundle** for execution after the appropriate monitoring action has taken place; and therefore enable the ICU to tackle problem based on state knowledge, e.g. completed state of an dispatched instructions (see col. 7, line 44 to col 7, line 17; col. 15, line 36 to col. 16, line 22) in a timely manner without delays that would have resulted in cache miss (see col. 16, line 59 to col. 17, line 67)

Final Office Action dated June 17, 2008 p. 7 (emphasis in original).

The Examiner bears the burden of establishing a *prima facie* case of obviousness based on prior art when rejecting claims under 35 U.S.C. § 103. *In re Fritch,* 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). The prior art reference (or references when combined) must teach or suggest all the claim limitations. *In re Royka,* 490 F.2d 981, 180 USPQ 580 (CCPA 1974). In determining obviousness, the scope and content of the prior art are... determined; differences between the prior art and the claims at issue are... ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background the obviousness or non-obviousness of the subject matter is determined. *Graham v. John Deere Co.,* 383 U.S. 1 (1966). "Often, it will be necessary for a court to look to interrelated teachings of multiple patents; the effects of demands known to the design community or present in the marketplace; and the

background knowledge possessed by a person having ordinary skill in the art, all in order to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue." *KSR Int'l. Co. v. Teleflex*, Inc., No. 04-1350 (U.S. Apr. 30, 2007). "Rejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness." *Id.* (citing *In re Kahn*, 441 F.3d 977, 988 (CA Fed. 2006)).

Claim 1, as amended, is as follows:

> 1.   A method in an instruction cache of a data processing system for monitoring execution of instructions, the method comprising:
> receiving a bundle at an instruction cache, the bundle containing at least one instruction slot, wherein the instruction slot contains an instruction;
> responsive to receiving the bundle, determining by the instruction cache whether the bundle contains an indicator within at least one spare bit of the at least one instruction slot, wherein the indicator identifies the instruction as one that is to be monitored by a performance monitor unit;
> responsive to a determination that the bundle contains the indicator within the at least one instruction slot, sending a signal by the instruction cache to a performance monitor unit, wherein upon receiving the signal, the performance monitor unit increments a counter associated with the instruction wherein the incrementing provides a count of a number of times the instruction is executed; and
> sending the bundle from the instruction cache to a functional unit for execution of the instruction.

Gover does not teach or suggest all the claim limitations. *Gover* discloses a performance monitoring unit wherein system states are saved within a control register, if the performance monitor is set to a "history mode." *Gover* does not disclose the claim 1 feature of "determining by the instruction cache whether the bundle contains an indicator within at least one spare bit of the at least one instruction slot."

This feature of claim 1 has been amended. Support for the feature can be found on pages 26-27 of the Specification, and in Figure 5. Therein, the Applicants state:

> Turning next to **Figure 5**, a diagram illustrating a bundle is depicted in accordance with the preferred embodiment of the present invention. Bundle **500** contains instruction slot **502**, instruction slot **504**, instruction slot **506** and template **508**. As illustrated, bundle **500** contains 128 bits. Each instruction slot contains 41 bits, and template **508** contains 5 bits. Template **508** is used to identify stops within the current bundle and to map instructions within the slots to different types of execution units.
> *Spare bits within bundle **500** are used to hold indicators of the present invention. For example, indicators **510**, **512**, and **514** are*

*located within instruction slots **502, 504,** and **506,** respectively. These indicators may take various forms and may take various sizes depending on the particular implementation.* Indicators may use a single bit or may use multiple bits. A single bit may be used to indicate that events are to be counted in response to execution of that instruction. Multiple bits may be used to identify a threshold, such as a number of processor or clock cycles for instruction execution that may pass before events should be counted. Further, these bits may even be used as a counter for a particular instruction. A similar use of fields may be used for indicators that mark data or memory locations.

Specification, pp. 26-27. Emphasis added.

In rejecting the unamended claim limitation, the Examiner cites the following passage from *Gover*:

> Referring again to FIG. 2, the entries of reorder buffer 76 are read by completion logic 80 and exception logic 82 of sequencer unit 18. In response to the "exception" fields of reorder buffer 76, exception logic 82 handles exceptions encountered during execution of dispatched instructions. In response to the "finished" fields and "exception" fields of reorder buffer 76, completion logic 80 indicates "completion" of instructions in order of their programmed sequence. Completion logic 80 indicates "completion" of an instruction if it satisfies the following conditions.
>
> Condition 1--The execution unit (to which the instruction is dispatched) finishes execution of the instruction (such that "finished"=1 in the instruction's associated entry in reorder buffer 76);
>
> Condition 2--No exceptions were encountered in connection with any stage of processing the instruction (such that "exception"=0 in the instruction's associated entry in reorder buffer 76); and
>
> Condition 3--Any previously dispatched instruction satisfies Condition 1 and Condition 2.
>
> In response to information in reorder buffer 76, dispatch logic 74 determines a suitable number of additional instructions to be dispatched.

Gover, col. 7, l. 62-col. 8, l. 17.

The cited passage discloses a methodology by which *Gover* keeps track of a sequence of instructions. The PowerPC architecture utilized by *Gover* provides for out of order execution of instructions, a relatively new phenomenon for microprocessors. This out of order execution allows the processor to avoid waiting for the completion of a cache miss by executing instructions out of sequence. However because instructions are often completed before a cache miss occurs, the PowerPC architecture must keep track of particular instruction branches, and which instructions have been completed.

Each instruction is allocated to an entry within the reorder buffer. A completion pointer and an allocation pointer mark are used to mark "finished" fields and locations where new entries for additional instructions can be pushed into the reorder buffer.

Nothing in the cited passage, or elsewhere in *Gover* discloses "determining by the instruction cache whether the bundle contains an indicator within at least one spare bit of the at least one instruction slot," as recited in amended claim 1.

The Examiner also points to Figures 5, 6a, and 6b as disclosing the claim 1 feature of "determining by the instruction cache whether the bundle contains an indicator within at least one spare bit of the at least one instruction slot." Figure 5 is a high level flowchart describing the implementation of a performance monitor within a data processing system. Figures 6a and 6b illustrate various bits within a monitor mode control register. Various actions by the performance monitor are controlled based on which bits of the monitor mode control register are set.

Again, nothing in any of Figures 5, 6a, and 6b discloses the claim 1 feature of "determining by the instruction cache whether the bundle contains an indicator within at least one spare bit of the at least one instruction slot." Because *Gover* does not disclose at least the claim 1 feature of "determining by the instruction cache whether the bundle contains an indicator within at least one spare bit of the at least one instruction slot," *Gover* cannot support a *prima facie* case of obviousness against claim 1 as amended. Therefore, the rejection of claim 1 under 35 U.S.C. § 103 has been overcome. Withdrawal of the rejection therefore requested.

Claims 6, 26, 29 and 30 depend from claim 1. The discussion of claim 1 above therefore applies to these dependent claims as well. By virtue of at least their dependence from claim 1, claims 6, 26, 29 and 30 are also not made obvious by *Gover*. In light of the reasons presented herein and the dependence of claims 6, 26, 29 and 30 from claim 1, *Gover* cannot support a *prima facie* case of obviousness against claims 6, 26, 29 and 30. Therefore, the rejection of claims 6, 26, 29 and 30 under 35 U.S.C. § 103 has been overcome. Withdrawal of the rejection is therefore requested.

Independent claims 32 and 41 recites features similar to those found in claim 1. Therefore the reasons presented above in regard to claim 1's patentability over *Gover* are equally applicable to claim 32. In light of the reasons presented above in regard to claim 1, *Gover* cannot support a *prima facie* case of obviousness against claims 32 and 41. Therefore, the rejection of claims 32 and 41 under 35 U.S.C. § 103 has been overcome. Withdrawal of the rejection is therefore requested.

Claims 34, 35, 38 and 39 depend from claim 31. By virtue of at least their dependence from claim 31, claims 34, 35, 38 and 39 are also not made obvious by *Gover*. In light of the

reasons presented herein and the dependence of claims 34, 35, 38 and 39 from their respective independent claim, *Gover* cannot support a *prima facie* case of obviousness against claims 34, 35, 38 and 39. Therefore, the rejection of claims 34, 35, 38 and 39 under 35 U.S.C. § 103 has been overcome. Withdrawal of the rejection is therefore requested.

IV.   Response to the Examiner's Assertions

IV.A.   First Erroneous Assertion:

On page 8 of the Office Action, the Examiner states the following:

> *Gover* does not explicitly disclose 'instruction cache' unit that *receives the bundle, determine the instruction to be counted based on the indicator for monitoring, and sending the bundle from the ICU to the execution unit;* that is, not implementing the instruction cache, the sequence unit (see Fig. 1) and the monitoring unit (Fig. 2; Fig. 4; col. 7, lines 15 to col. 7, lines 17) in a combined functional unit such as a single instruction cache unit (ICU).
>
> …
>
> The meta-information being dispatched from the bundle of instructions coming from the reorder buffer, sequencer, rename buffer, and the tight relationship thereof with the original instruction cache (Fig. 1) and the performance monitor by *Gover entails* that the dispatched bundle contains instructions or data back and forth between cache and hardware monitoring tool.

Final Office Action dated June 17, 2008 p. 7

The Examiner admits to several deficiencies of *Gover*, but then concludes that the bundle received by the instruction cache surely must contain an indicator for monitoring, based on the litany of other features disclosed by *Gover*. Such a conclusion by the Examiner is manifestly false.

While the Examiner states that "*Gover entails* that the dispatched bundle contains instructions or data back and forth between cache and hardware monitoring tool," the Examiner fails to cite anything actually indicating such. As stated above, the Examiner instead recites a variety of features, and then makes a conclusion that is wholly unsupported by the Examiner's presuppositions.

The cited reorder buffer, sequencer and rename buffer are all utilized to maintain an order of execution of instructions within the PowerPC architecture of *Gover*. Despite the Examiner's assertions to the contrary, the reorder buffer, sequencer and rename buffer are all only loosely associated with the hardware monitoring tool.

Furthermore, the Examiner's statement that the dispatched bundle contains instructions or data back and forth between cache and hardware monitoring tool is contrary to the language of

amended claim 1. The applicant is not claiming that instructions are sent from the instruction cache to the performance monitor. Rather, claim 1 recites that, "responsive to a determination that the bundle contains the indicator within the at least one instruction slot, *sending a signal by the instruction cache to a performance monitor unit*...." That is, even if the Examiner is correct regarding the purported "tight relationship," a point that the Applicants dispute, such a conclusion is completely irrelevant in light of the language of claim 1. Applicants are not claiming that an instruction is communicated between the instruction cache and the performance monitor.

IV.B. Second Erroneous Assertion:

On page 14 of the Office Action, the Examiner explains the §112/ new matter rejection. The Examiner has objected to the instruction cache performing any operation, and instead has adopted the ultra-narrow view that the cache is simply a repository for temporary information. As stated in a previous response, an instruction cache generally consists of the cache memory location, i.e., what the Examiner narrowly refers to as the cache, as well as a cache controller. However, both of these are needed to comprise "a cache." While any determination of the contents of instructions may technically be performed by the cache controller, the determination is still performed by "the cache." The Applicants submit that the Examiner's definition of "a cache" is not consistent with the general usage of the term by one of ordinary skill in the art. Cache controllers are well known and enabled in the art, and thus the Specification, as written, is consistent with the known concept of "a cache."

A patent specification need not teach, and preferably omits, what is well known in the art. *Hybritech Inc. v. Monoclonal Antibodies, Inc.*, 802 F.2d 1367, 1384, 231 USPQ 81, 94 (Fed. Cir. 1986). Many high performance computer architectures, such as the PowerPC, preemptively examine and process instructions within the cache. This preemptive examination requires some level of pre-processing of information by the cache, prior to the instruction being sent to a processor for execution. This can include branch determination, prerequisite instruction completion, etc. Each of these processes can be completed by a cache controller logic implemented within the cache. Thus, the cache performs the various actions.

The applicants are not under any duty to enable what is not novel. Thus, the Applicants submit that the Application as originally filed is sufficient to enable the processing steps occurring in the instruction cache. These process steps are delineated in Figure 7.

Furthermore, the Examiner concludes that the Applicants should provide "concrete evidence that any cache subsystem inherently includes a cache controller having itself a determination functionality." This is an impermissible attempt to switch the burden of proof to

the Applicants. A determination of inherency is one that the Examiner is required to make, not the Applicants. The law is clear on this. If the Examiner believes that a cache controller is inherently NOT included in an instruction cache system, the Examiner should state such. However, preemptively requiring the Applicants to submit "concrete evidence that any cache subsystem inherently includes a cache controller" impermissibly shifts the burden to the Applicant.

V.   Conclusion

It is respectfully urged that the subject application is patentable over the cited references and is now in condition for allowance.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: September 23, 2008

Respectfully submitted,

/Brandon G. Williams/

Brandon G. Williams
Reg. No. 25,035
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicants